下载APP

# 10 | 设置工作模式与环境（上）：建立计算机

2021-05-31 LMOS

《操作系统实战45讲》 课程介绍 >

**讲述：陈晨**

时长 11:42 大小 10.73M

你好，我是 LMOS。

经过前面那么多课程的准备，现在我们距离把我们自己操作系统跑起来，已经是一步之遥了。现在，你是不是很兴奋，很激动？有这些情绪说明你是喜欢这门课程的。

接下来的三节课，我们会一起完成一个壮举，从 GRUB 老大哥手中接过权柄，让计算机回归到我们的革命路线上来，为我们之后的开发自己的操作系统做好准备。

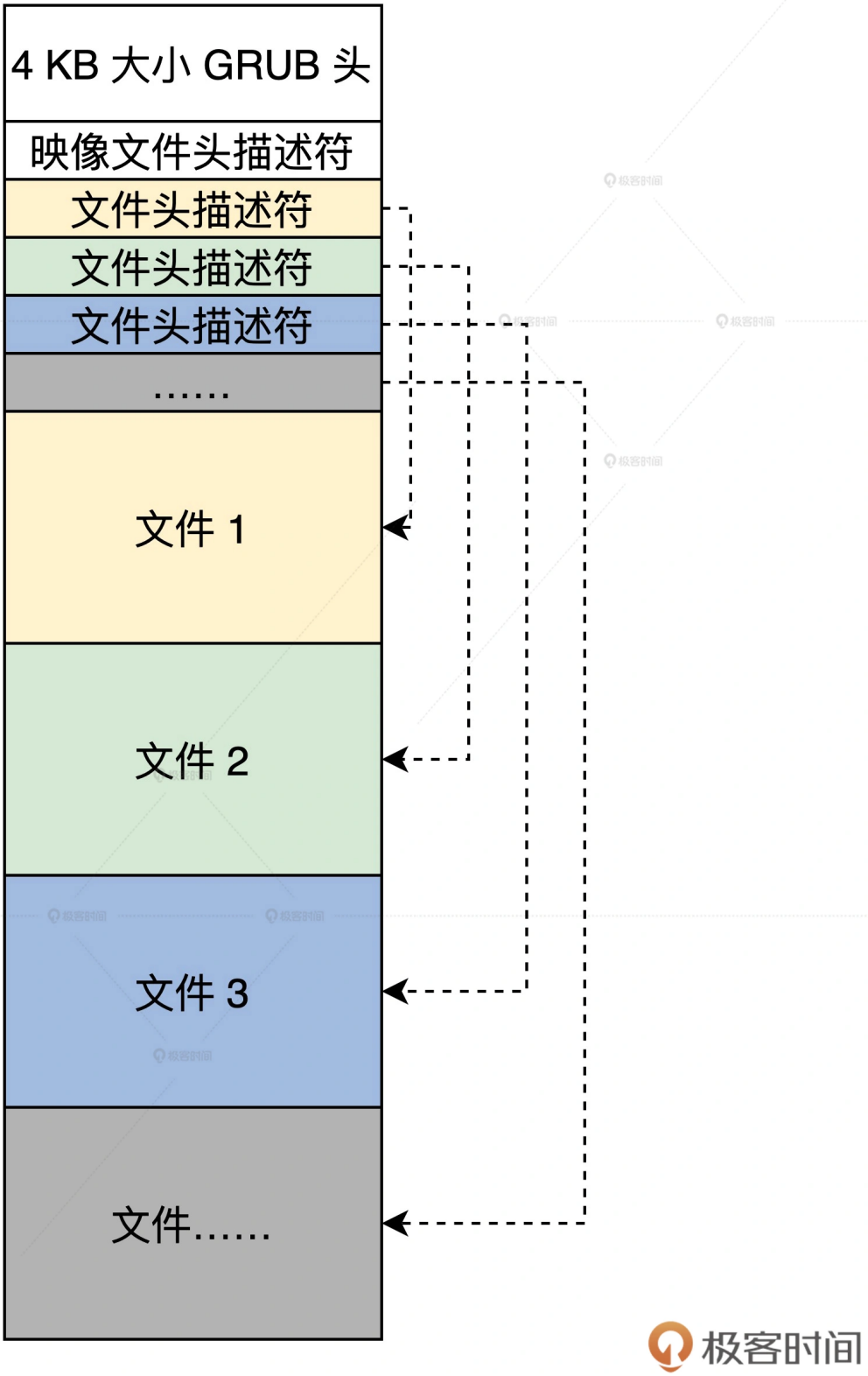
具体我是这样来安排的，今天这节课，我们先来搭好操作系统的测试环境。第二节课，我们一起实现一个初始化环境的组件——二级引导器，让它真正继承 GRUB 权力。第三节课，我们正式攻下初始化的第一个山头，对硬件抽象层进行初始化。

好，让我们正式开始今天的学习。首先我们来解决内核文件封装的问题，然后动手一步步建好虚拟机和生产虚拟硬盘。课程配套代码你可以在这里下载。

## 从内核映像格式说起

我们都知道，一个内核工程肯定有多个文件组成，为了不让 GRUB 老哥加载多个文件，因疲劳过度而产生问题，我们决定让 GRUB 只加载一个文件。

但是要把多个文件变成一个文件就需要封装，即把多个文件组装在一起形成一个文件。这个文件我们称为**内核映像文件**，其中包含二级引导器的模块，内核模块，图片和字库文件。为了这映像文件能被 GRUB 加载，并让它自身能够解析其中的内容，我们就要定义好具体的格式。如下图所示。




内核映像文件格式

上图中的 GRUB 头有 4KB 大小，GRUB 正是通过这一小段代码，来识别映像文件的。另外，根据映像文件头描述符和文件头描述符里的信息，这一小段代码还可以解析映像文件中的其它文件。

映像文件头描述符和文件描述符是两个 C 语言结构体，如下所示。

复制代码

```
//映像文件头描述符
typedef struct s_mlosrddsc
{
    u64_t mdc_mgic; //映像文件标识
    u64_t mdc_sfsum;//未使用
    u64_t mdc_sfsoff;//未使用
    u64_t mdc_sfeoff;//未使用
    u64_t mdc_sfrlsz;//未使用
    u64_t mdc_ldrbk_s;//映像文件中二级引导器的开始偏移
    u64_t mdc_ldrbk_e;//映像文件中二级引导器的结束偏移
    u64_t mdc_ldrbk_rsz;//映像文件中二级引导器的实际大小
    u64_t mdc_ldrbk_sum;//映像文件中二级引导器的校验和
    u64_t mdc_fhdbk_s;//映像文件中文件头描述的开始偏移
    u64_t mdc_fhdbk_e;//映像文件中文件头描述的结束偏移
    u64_t mdc_fhdbk_rsz;//映像文件中文件头描述的实际大小
    u64_t mdc_fhdbk_sum;//映像文件中文件头描述的校验和
    u64_t mdc_filbk_s;//映像文件中文件数据的开始偏移
    u64_t mdc_filbk_e;//映像文件中文件数据的结束偏移
    u64_t mdc_filbk_rsz;//映像文件中文件数据的实际大小
    u64_t mdc_filbk_sum;//映像文件中文件数据的校验和
    u64_t mdc_ldrcodenr;//映像文件中二级引导器的文件头描述符的索引号
    u64_t mdc_fhdnr;//映像文件中文件头描述符有多少个
    u64_t mdc_filnr;//映像文件中文件头有多少个
    u64_t mdc_endgic;//映像文件结束标识
    u64_t mdc_rv;//映像文件版本
}mlosrddsc_t;

#define FHDSC_NMAX 192 //文件名长度
//文件头描述符
typedef struct s_fhdsc
{
    u64_t fhd_type;//文件类型
    u64_t fhd_subtype;//文件子类型
    u64_t fhd_stuts;//文件状态
    u64_t fhd_id;//文件id
    u64_t fhd_intsfsoff;//文件在映像文件位置开始偏移
    u64_t fhd_intsfend;//文件在映像文件的结束偏移
    u64_t fhd_frealsz;//文件实际大小
    u64_t fhd_fsum;//文件校验和
    char   fhd_name[FHDSC_NMAX];//文件名
}fhdsc_t;
```

有了映像文件格式，我们还要有个打包映像的工具，我给你提供了一个 Linux 命令行下的工具，你只要明白使用方法就可以，如下所示。

复制代码

```
lmoskrlimg -m k -lhf GRUB头文件 -o 映像文件 -f 输入的文件列表
-m 表示模式 只能是k内核模式
-lhf 表示后面跟上GRUB头文件
-o 表示输出的映像文件名
-f 表示输入文件列表
例如：lmoskrlimg -m k -lhf grubhead.bin -o kernel.img -f file1.bin file2.bin fi
```

## 准备虚拟机

打包好了映像文件，我们还有很重要的一步配置——准备虚拟机。这里你不妨先想一想，开发应用跟开发操作系统有什么不同呢？

在你开发应用程序时，可以在 IDE 中随时编译运行应用程序，然后观察结果状态是否正确，中间可能还要百度一下查找相关资料，不要笑，这是大多数人的开发日常。但是你开发操作系统时，不可能写 5 行代码之后就安装在计算机上，重启计算机去观察运行结果，这非常繁琐，也很浪费时间。

好在我们有虚拟机这个好帮手。虚拟机用软件的方式实现了真实计算机的全部功能特性，它在我们所使用的 Linux 下，其实就是个应用程序。

使用虚拟机软件我们就可以在现有的 Linux 系统之上开发、编译、运行我们的操作系统了，省时且方便。节约的时间我们可以喝茶、听听音乐、享受美好生活。

### 安装虚拟机

这里我们一致约定**使用甲骨文公司的**VirtualBox**虚拟机**。经过测试，我发现 VirtualBox 虚拟机有很多优点，它的功能相对完善、性能强、BUG 少，而且比较稳定。

在现代 Linux 系统上安装 VirtualBox 虚拟机是非常简单的，你只要在 Linux 发行版中找到其应用商店，在其中搜索 VirtualBox 就行了。我们作为专业人士一条命令可以解决的事情，为什么要用鼠标点来点去呢，多浪费时间。

所以，你只要在终端中输入如下命令就行了，我假定你安装了 Ubuntu 系的 Linux 发行版，这里 Ubuntu 的版本不做规定。

复制代码

```
sudo apt-get install virtualbox-6.1
```

运行 Virtualbox 后，如果出现如下界面，就说明安装 VirtualBox 成功了。

安装VirtualBox

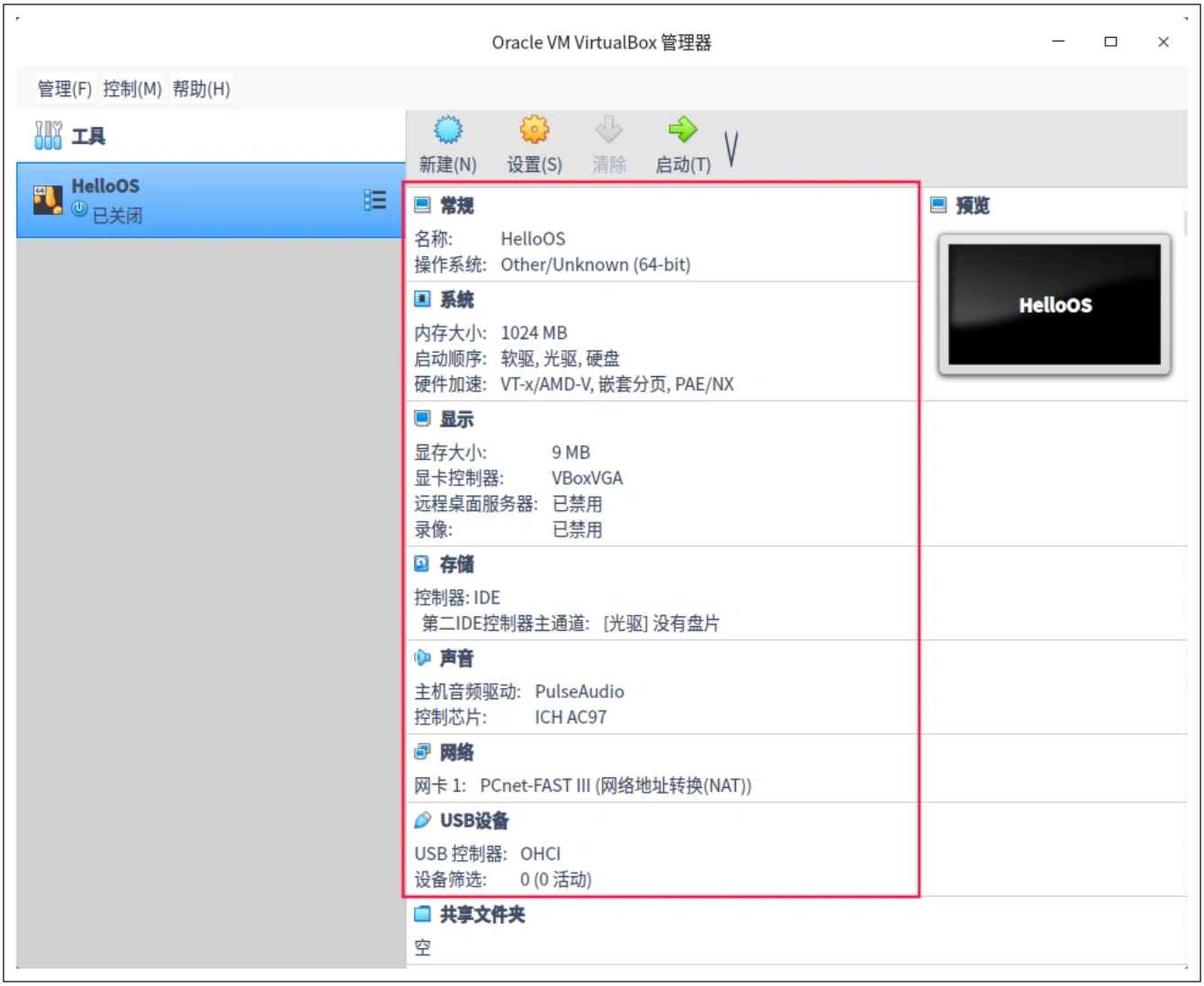
## 建立虚拟电脑

前面我们只是安好了虚拟机管理软件，我们还要新建虚拟机才可以。点击上图中的新建，然后选择专家模式，就可以进入专家模式配置我们的电脑了。

尽管它是虚拟的，我们还是可以选择 CPU 类型、内存大小、硬盘大小、网络等配置，为了一致性，请你按照如下截图来配置。

新建虚拟电脑
虚拟电脑名称和系统类型
名称: HelloOS
文件夹: /home/lmos/VirtualBox VMs
类型(T): Other
版本(V): Other/Unknown (64-bit)
内存大小(M)
)24
MB
4 MB
12288 MB
内存：1024MB
虚拟硬盘
不添加虚拟硬盘(D)
现在创建虚拟硬盘(C)
使用已有的虚拟硬盘文件(U)
没有盘片
向导模式(G)
< 上一步(B)
创建
取消

新建虚拟机




新建虚拟机

可以看到，我们选择了 64 位的架构，1024MB 内存，但是不要添加硬盘，后面自有妙用。显卡是 VBoxVGA，还有硬件加速，这会让虚拟机调用我们机器上真实的 CPU 来运行我们的操作系统。

## 手工生产硬盘

上面的虚拟机中还没有硬盘，没有硬盘虚拟机就没地方加载数据，我们当然不是要买一块硬盘挂上去，而是要去手工生产一块硬盘。你马上就会发现，从零开始生产一块虚拟硬盘，这比从零开始写一个操作系统简单得多。

至于为什么手工生产硬盘，我先卖个关子，你看完这部分内容就能找到答案。

其实大多数虚拟机都是用文件来模拟硬盘的，即主机系统（HOST OS 即你使用的物理机系统 ）下特定格式的文件，虚拟机中操作系统的数据只是写入了这个文件中。

## 生产虚拟硬盘

其实虚拟机只是用特定格式的文件来模拟硬盘，所以生产虚拟硬盘就变成了生成对应格式的文件，这就容易多了。我们要建立 100MB 的硬盘，这意味着要生成 100MB 的大文件。

下面我们用 Linux 下的 **dd 命令**（用指定大小的块拷贝一个文件，并在拷贝的同时进行指定的转换）生成 100MB 的纯二进制的文件（就是 1～100M 字节的文件里面填充为 0），如下所示。

复制代码

```
dd bs=512 if=/dev/zero of=hd.img count=204800

;bs:表示块大小，这里是512字节
;if：表示输入文件，/dev/zero就是Linux下专门返回0数据的设备文件，读取它就返回0
;of：表示输出文件，即我们的硬盘文件。
;count：表示输出多少块
```

执行以上命令就可以生成 100MB 的文件。文件数据为全 0。由于我们不用转换数据，就是需要全 0 的文件，所以 dd 命令只需要这几个参数就行。

## 格式化虚拟硬盘

虚拟硬盘也需要格式化才能使用，所谓格式化就是在硬盘上建立文件系统。只有建立了文件系统，现有的成熟操作系统才能在其中存放数据。

可是，问题来了。虚拟硬盘毕竟是个文件，如何让 Linux 在一个文件上建立文件系统呢？这个问题我们要分成三步来解决。

第一步，把虚拟硬盘文件变成 Linux 下的回环设备，让 Linux 以为这是个设备。其实在 Linux 下文件可以是设备，设备可以是文件。下面我们用 losetup 命令，将 hd.img 变成 Linux 的回环设备，代码如下。

复制代码

```
sudo losetup /dev/loop0 hd.img
```

第二步，将 losetup 命令用于设置回环设备。回环设备可以把文件虚拟成 Linux 块设备，用来模拟整个文件系统，让用户可以将其看作硬盘、光驱或软驱等设备，并且可用 mount 命令挂载当作目录来使用。

我们可以用 Linux 下的 mkfs.ext4 命令格式化这个 /dev/loop0 回环块设备，在里面建立 EXT4 文件系统。

复制代码

```
sudo mkfs.ext4 -q /dev/loop0
```

第三步，我们用 Linux 下的 mount 命令，将 hd.img 文件当作块设备，把它挂载到事先建立的 hdisk 目录下，并在其中建立一个 boot，这也是后面安装 GRUB 需要的。如果能建立成功，就说明前面的工作都正确完成了。

说到这里，也许你已经想到了我们要手工生成硬盘的原因。**这是因为 mount 命令只能识别在纯二进制文件上建立的文件系统，如果使用虚拟机自己生成的硬盘文件，mount 就无法识别我们的文件系统了。**

复制代码

```
sudo mount -o loop ./hd.img ./hdisk/ ;挂载硬盘文件
sudo mkdir ./hdisk/boot/ ;建立boot目录
```

进行到这里，我们会发现 hdisk 目录下多了一个 boot 目录，这说明我们挂载成功了。

## 安装 GRUB

正常安装系统的情况下，Linux 会把 GRUB 安装在我们的物理硬盘上，可是我们现在要把 GRUB 安装在我们的虚拟硬盘上，而且我们的操作系统还没有安装程序。所以，我们得利用一下手上 Linux（HOST OS），通过 GRUB 的安装程序，把 GRUB 安装到指定的设备上（虚拟硬盘）。

想要安装 GRUB 也不难，具体分为两步，如下所示。

复制代码

```
第一步挂载虚拟硬盘文件为loop0回环设备
sudo losetup /dev/loop0 hd.img
sudo mount -o loop ./hd.img ./hdisk/ ;挂载硬盘文件
第二步安装GRUB
sudo grub-install --boot-directory=./hdisk/boot/ --force --allow-floppy /dev/l
；--boot-directory 指向先前我们在虚拟硬盘中建立的boot目录。
：--force --allow-floppy ：指向我们的虚拟硬盘设备文件/dev/loop0
```

可以看到，现在 /hdisk/boot/ 目录下多了一个 grub 目录，表示我们的 GRUB 安装成功。请注意，这里还要在 /hdisk/boot/grub/ 目录下建立一个 **grub.cfg 文本文件**，GRUB 正是通过这个文件内容，查找到我们的操作系统映像文件的。

我们需要在这个文件里写入如下内容。

复制代码

```
menuentry 'HelloOS' {
insmod part_msdos
insmod ext2
set root='hd0,msdos1' #我们的硬盘只有一个分区所以是'hd0,msdos1'
multiboot2 /boot/HelloOS.eki #加载boot目录下的HelloOS.eki文件
boot #引导启动
}
set timeout_style=menu
if [ "${timeout}" = 0 ]; then
  set timeout=10 #等待10秒钟自动启动
fi
```

## 转换虚拟硬盘格式

你可能会好奇，我们前面好不容易生产了 mount 命令能识别的虚拟硬盘，这里为什么又要转换虚拟硬盘的格式呢？

这是因为这个纯二进制格式只能被我们使用的 Linux 系统识别，但不能被虚拟机本身识别，但是我们最终目的却是让这个虚拟机加载这个虚拟硬盘，从而启动其中的由我们开发的操作系统。

好在虚拟机提供了专用的转换格式的工具，我们只要输入一行命令即可。

复制代码

```
VBoxManage convertfromraw ./hd.img --format VDI ./hd.vdi
```

```
;convertfromraw 指向原始格式文件
；--format VDI  表示转换成虚拟需要的VDI格式
```

## 安装虚拟硬盘

好了，到这里我们已经生成了 VDI 格式的虚拟硬盘，这正是我们虚拟机所需要的。然而虚拟硬盘必须要安装虚拟机才可以运行，也就是这个 hd.vdi 文件要和虚拟机软件联系起来。

因为我们之前在建立虚拟机时并没有配置硬盘相关的信息，所以这里需要我们进行手工配置。

配置虚拟硬盘分两步：第一步，配置硬盘控制器，我们使用 SATA 的硬盘，其控制器是 intelAHCI；第二步，挂载虚拟硬盘文件。

具体操作如下所示。

复制代码

```
#第一步 SATA的硬盘其控制器是intelAHCI
VBoxManage storagectl HelloOS --name "SATA" --add sata --controller IntelAhci
#第二步
VBoxManage closemedium disk ./hd.vdi #删除虚拟硬盘UUID并重新分配
#将虚拟硬盘挂到虚拟机的硬盘控制器
VBoxManage storageattach HelloOS --storagectl "SATA" --port 1 --device 0 --typ
```

因为 VirtualBox 虚拟机用 UUID 管理硬盘，所以每次挂载硬盘时，都需要删除虚拟硬盘的 UUID 并重新分配。

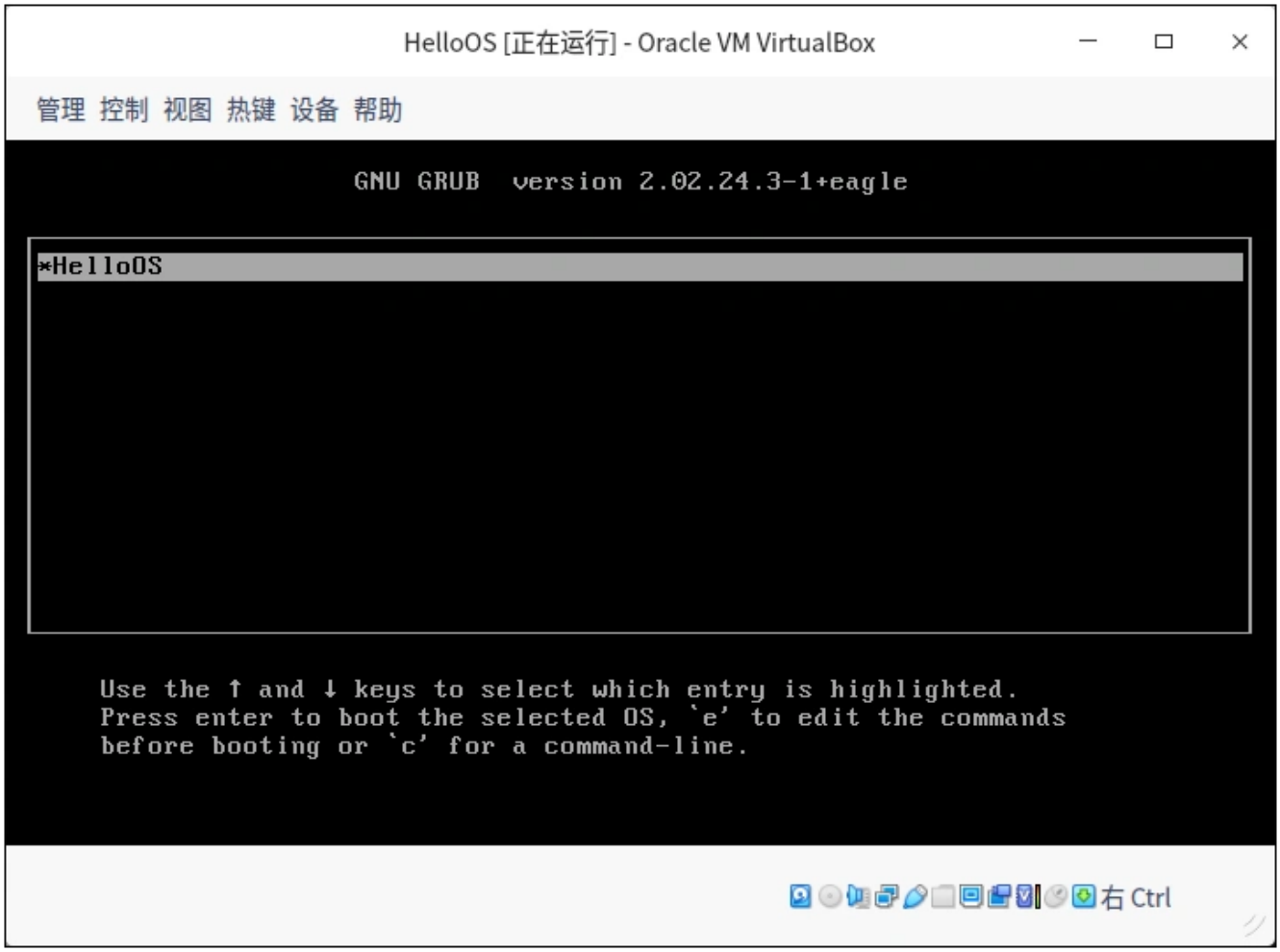
## 最成功的失败

现在硬盘也安装好了，下面终于可以启动我们的虚拟电脑了，我们依然通过命令启动，在 Linux 终端中输入如下命令就可以了。

复制代码

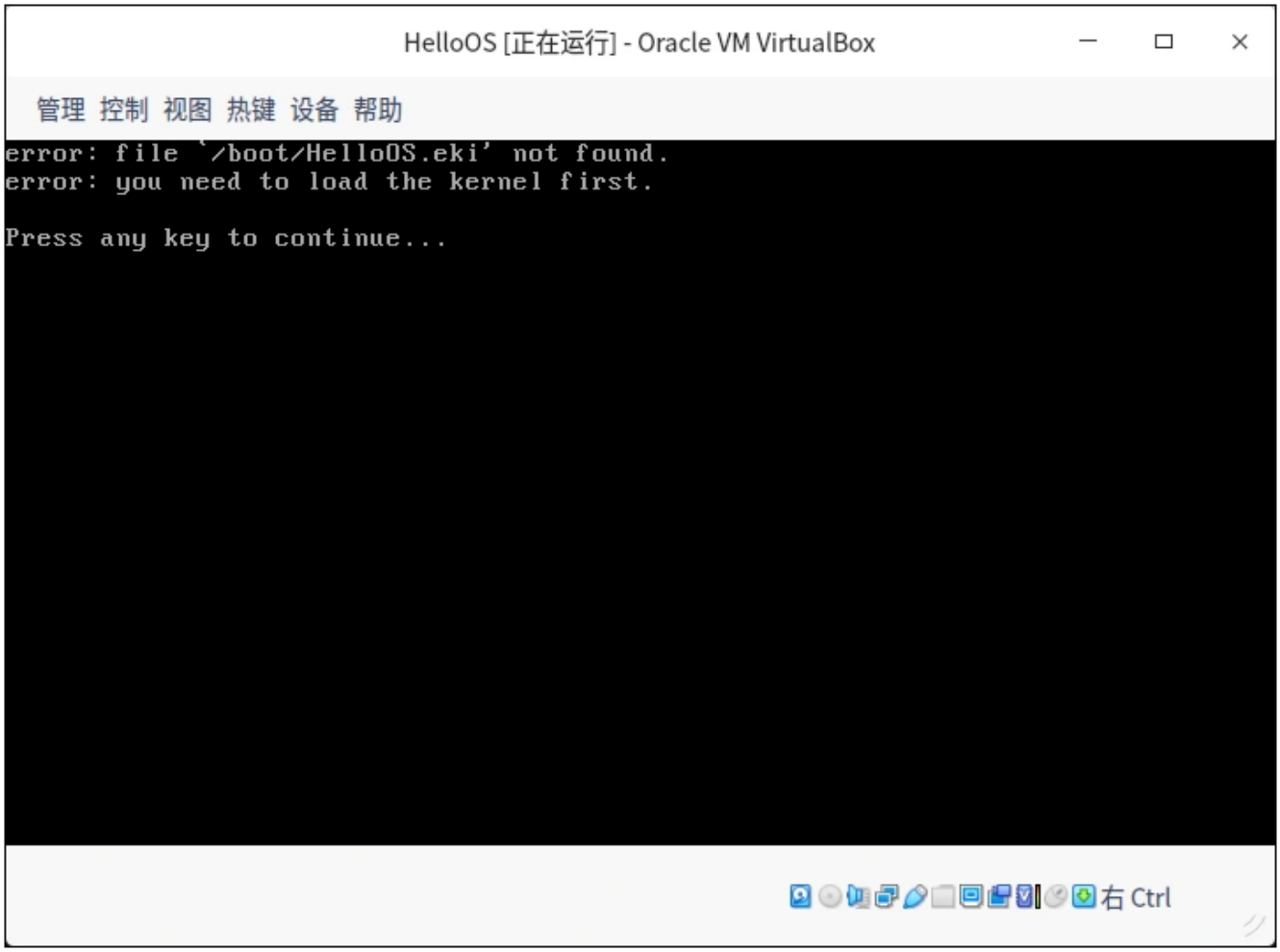
```
VBoxManage startvm HelloOS #启动虚拟机
```

输入以上命令就会出现以下界面，出现 GRUB 引导菜单。




虚拟机启动

直接按下回车键，就能选择我们的 HelloOS，GRUB 就会加载我们的 HelloOS，但是会出现如下错误。




虚拟机GRUB未找到文件

上面的错误显示，GRUB 没有找到 HelloOS.eki 文件，这是因为我们从来没有向虚拟硬盘中放入 HelloOS.eki 文件，所以才会失败。

**但这是我们最成功的失败，因为我们配置好了虚拟机，手动建造了硬盘，并在其上安装了 GRUB，到这里我们运行测试环境已经准备好了。**

其实你不必太过担心，等我们完成了二级引导器的时候，这个问题会迎刃而解。

## 重点回顾

希望今天这节课给你带来成就感，虽然我们才走出了万里长征的第一步。为了这一步我们准备了很多。但是我们始终没忘记这一课程的目的，即我们要从 GRUB 老大哥手里接过权柄，控制计算机王国，为此，我们完成了后面这三个工作。

1. 我们了解了内核映像格式，以便我们对编译产生的内核程序文件进行封装打包。
2. 为了方便测试我们的操作系统，我们了解并安装了虚拟机。

3. 手动建立了虚拟硬盘，对其格式化，在其中手动安装了 GRUB 引导器，并且启动了虚拟电脑。

虽然我们启动虚拟电脑失败了，但是对我们而言却是巨大的成功，因为它标志着我们测试运行内核的环境已经成功建立，下一课我们将继续实现二级引导器。

## 思考题

请问，我们为什么要把虚拟硬盘格式化成 ext4 文件系统格式呢？

欢迎你在留言区跟我交流探讨，如果你身边有对写操作系统感兴趣的朋友，也欢迎把这节课分享给他，一起学习。

好，我是 LMOS，我们下节课见！

分享给需要的人，Ta订阅后你可得 20 元现金奖励

赞 9 提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。 页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 09 | 瞧一瞧Linux：Linux的自旋锁和信号量如何实现？

下一篇 11 | 设置工作模式与环境（中）：建造二级引导器

## 精选留言 (35)

写留言

**第九天魔王** 置顶
2021-05-31

已完成实验，有几个心得分享一下。
1，如果主机就是Linux比如Ubuntu，则完全按照东哥的教程做即可，即Linux主机安装虚拟机启动HelloOS。
2，如果主机是Windows，启动虚拟机运行Ubuntu，很多童鞋担心虚拟机Ubuntu里再运行虚拟机启动HelloOS，这样就会套娃。其实可以避免套娃的。在你的虚拟机，也就是U...
展开

编辑回复: 6666。十分感谢你的分享，已置顶供后面同学参考。

8 38

**neohope** 置顶
2021-06-01

【更新】对于使用Windows、MacOS的同学，或不熟悉命令行的同学可以这样操作【不需要套娃，也不用做硬盘格式转换】：
1、先安装一个Linux虚拟机A，比如Ubuntu18或Ubuntu20
2、关闭虚拟机，在这个虚拟机上，用向导添加一个新硬盘hd.vdi
3、启动虚拟机A...

展开

作者回复: 谢谢你的分享，让更多的同学明白了 如何 建立 环境

9 23

**Veek** 置顶
2021-06-05

终于试成功了，真开心，断断续续搞了快一个星期。
因为我的主机是mac, 谢谢第九天魔王大、neohoped大和老王大的分享，免去套娃，少走很多弯路。

0.在主机Mac上安装 Virtual Box 6.1，开了2个虚拟机：...
展开

作者回复: 你好，你的经验可以多多分享

11 15

**springXu** 置顶
2021-05-31

这节课是环境搭建的操作步骤，还没开始真正的动手写代码的要求。
这些操作首先生成内核文件kernel.img文件，这个文件应该是启动的文件。猜测把这个改成HelloOS.eki文件放到grub配置的目录下，不会出现not found的提示。至于显示什么操作就知道了。
二级引导器应该是initldrimh.bin文件的内容。 这部分内容估计是下节课要讲节的内容...
展开

作者回复: 用文件系统 是为了linux可以像其中写入文件（我们的内核）

2 5

**Veek** 置顶
2021-06-05

超过2000字，补发
后记：

卡了很久，可以看到GRUB，但看不到*HelloOS

...

展开

作者回复: 你好，这个警告可以忽略

4

**Shinymimiq** 置顶

2021-06-11

我来优化一下Mac小伙伴的开发流程，直接能用Mac编译，并修改我们Cosmos的硬盘镜像文件了，基本可以不需要启动Ubuntu虚拟机了。

首先得按评论区里Veek或者Neohoped大佬的教程得到一个hd.vdi之类的镜像文件

...

展开

作者回复: 你好，这个厉害了

1 3

**老王** 置顶

2021-06-01

在整个操作中
安装GRUB这一步：
sudo grub-install --boot-directory=./hdisk/boot/ --force --allow-floppy /dev/loop0
在ubutnu16.04下会报告一下问题...

展开

作者回复: 是的

4 2

**青玉白露** 置顶

2021-07-15

参考neohope的方法，其实在Mac和Windows环境下利用VirtualBox可以很方便的进行测试，
只需要建立两个虚拟机，一个ubuntu一个是我们的COSMOS，自己在Mac上进行了测

试，具体的细节可以参考博客：
https://blog.csdn.net/ll15982534415/article/details/118771286

展开

作者回复: 哈哈 阔以

1

**pedro**
2021-05-31

一大早起来第一件事就是看专栏！
今天专栏实操性是很强的，容易上手，产生成就感。
我昨天就clone代码跑起来了，对于virtualbox大家如果觉得命令行难用，可以直接打开界面点击按钮新建虚拟机，对于追求geek的人，命令行还是得来一次。
思考题：...

展开

作者回复: 铁子 回答的相当正确

10

**lunar**
2021-06-02

很开心，我已经成功的跟不上了

展开

编辑回复: 慢慢来，别灰心。

5

**Fan**
2021-06-07

losetup -f 会返回第一个未使用的环回设备，比如我这里返回的就是/dev/loop0
losetup -a 会返回当前已经使用的所有环回设备；

展开

作者回复: 6666，谢谢分享

3

**尼欧**
2021-05-31

打卡

展开

编辑回复: 打卡成功，学习愉快！

3

**Zhendicai**
2021-06-14

所以手工制作虚拟硬盘的主要原因是为了向虚拟硬盘中安装GRUB？ 中间转换硬盘格式是因为mount不能挂载由虚拟机直接产生的硬盘从而导致不能直接利用Linux写入GRUB？那这是说明虚拟机创建的硬盘中是不带有GRUB的么？

作者回复: 是的啊

2

**Json**
2021-05-31

同学们，宿主机是windows咋办？不想整套娃。。。

展开

编辑回复: 可以参考一下第九天魔王同学的分享，可以不套娃。

3 2

**Stephen**
2021-06-27

执行~$ VBoxManage storagectl HelloOS --name "SATA" --add sata --controller IntelAhci --portcount 1
报出如下错误:
VBoxManage: error: Could not find a registered machine named 'HelloOS'
VBoxManage: error: Details: code VBOX_E_OBJECT_NOT_FOUND (0x80bb0001), c...

展开

作者回复: 你好，要先建立一个名为HelloOS的虚拟电脑

2 1

**陶建**
2021-06-22

1. grub-install 设置为i386-pc方式, efi方式启动出现问题. 指定 --target=i386-pc
2. loop设备的操作
sudo losetup /dev/loop5 ./hd.img
sudo mount -o loop ./hd.img ./hdisk
sudo umount ./hdisk...

展开

作者回复: 可以的

1 1

**Stephen**
2021-06-21

这里 sudo apt-get install virtualbox-6.1会有以下错误：
E: Couldn't find any package by glob 'virtualbox-6.1'
E: Couldn't find any package by regex 'virtualbox-6.1'
这里我采用stackofflow stackexample 的答案：
echo "deb http://download.virtualbox.org/virtualbox/debian $(lsb_release -cs) co...

展开

作者回复: 66666

1

**Fan**
2021-06-06

请问，我们为什么要把虚拟硬盘格式化成 ext4 文件系统格式呢？
其他格式的文件系统格式可以吗？

展开

作者回复: 可以的 只要Linux和GRUB，能识别的 都可以

1

**Hezhen**
2021-06-03

我按照步骤，最后启动HelloOS进入了 grub>界面，和文档中不一样，请问这怎么弄啊

1 1

**Victor**
2021-06-01

我在ubuntu 16.04上做实验，安装GRUB时报错：
# grub-install --boot-directory=./hdisk/boot/ --force --allow-floppy /dev/loop0
Installing for i386-pc platform.
grub-install: warning: File system `ext2' doesn't support embedding.
grub-install: error: embedding is not possible, but this is required for cross-disk i...
展开

编辑回复: 参考一下置顶几条实验记录，看看是不是版本问题或者loop资源问题。

2 1